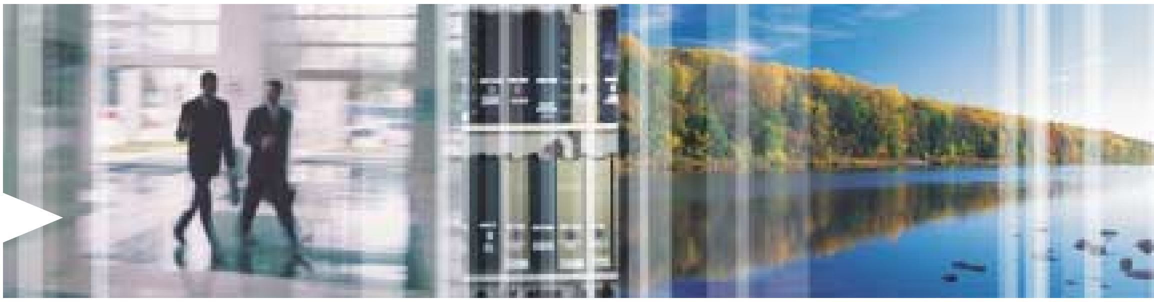
WWW.BMC.COM

CONTROL-D®
CA-DISPATCH
Conversion Guide

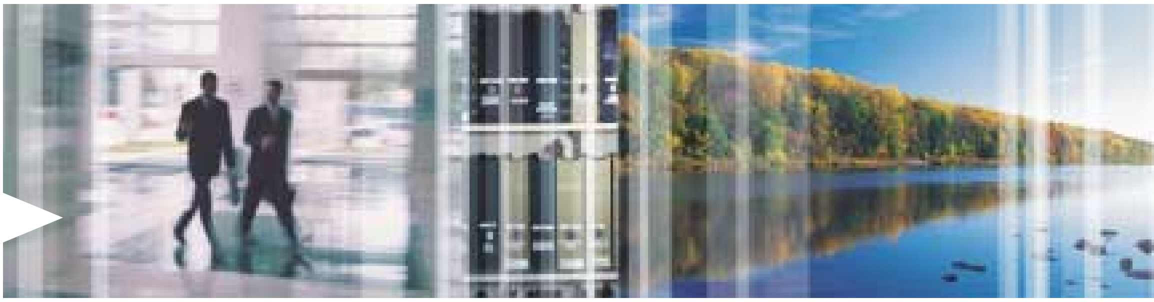


Supporting

CONTROL-D version 6.2.18

July 31, 2006

WWW.BMC.COM



Supporting

Contacting BMC Software

You can access the BMC Software website at http://www.bmc.com. From this website, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

United States and Canada

Address	BMC SOFTWARE INC 2101 CITYWEST BLVD HOUSTON TX 77042-2827 USA	**Telephone**	713 918 8800 or 800 841 2031	**Fax**	713 918 8000

Outside United States and Canada

Telephone	(01) 713 918 8800	**Fax**	(01) 713 918 8000

Restricted rights legend

Customer support

You can obtain technical support by using the Support page on the BMC Software website or by contacting Customer Support by telephone or e-mail. To expedite your inquiry, please see “Before Contacting BMC Software.”

Support website

You can obtain technical support from BMC Software 24 hours a day, 7 days a week at http://www.bmc.com/support_home. From this website, you can

- read overviews about support services and programs that BMC Software offers
- find the most current information about BMC Software products
- search a database for problems similar to yours and possible solutions
- order or download product documentation
- report a problem or ask a question
- subscribe to receive e-mail notices when new product versions are released
- find worldwide BMC Software support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by telephone or e-mail

In the United States and Canada, if you need technical support and do not have access to the web, call 800 537 1813 or send an e-mail message to support@bmc.com. Outside the United States and Canada, contact your local support center for assistance.

Before contacting BMC Software

Before you contact BMC Software, have the following information available so that Customer Support can begin working on your problem immediately:

- product information
  - product name
  - product version (release number)
  - license number and password (trial or permanent)
- operating system and environment information
  - machine type
  - operating system type, version, and service pack or other maintenance level such as PUT or PTF
  - system hardware configuration
  - serial numbers
  - related software (database, application, and communication) including type, version, and service pack or maintenance level
- sequence of events leading to the problem
- commands and options that you used
- messages received (and the time and date that you received them)
  - product error messages
  - messages from the operating system, such as `file system full`
  - messages from related software

Contents

Figures

Tables

About This Guide

This guide contains the information necessary to help you to convert from CA-DISPATCH to CONTROL-D. The guide contains the following parts:

Chapter 1 – Overview

Provides an introduction, outlines conversion steps, gives naming conventions, and gives a short explanation of each step of the conversion process.

Chapter 2 – Conversion Steps

Specifies the steps needed to perform the conversion.

Chapter 3 – Building a CONTROL-D Recipient Tree

Gives the procedure and examples for defining and building a CONTROL-D Recipient Tree using utility CTDBLDTR.

Appendix A – Default Conversion Parameters

Appendix B – Messages

Conventions Used in This Guide

Notational conventions that may be used in this guide are explained below.

Standard Keyboard Keys

Keys that appear on the standard keyboard are identified in boldface, for example, **Enter, Shift, Ctrl+S** (a key combination), or **Ctrl S** (a key sequence).

WARNING

The commands, instructions, procedures, and syntax illustrated in this guide presume that the keyboards at your site are mapped in accordance with the EBCDIC character set. Certain special characters are referred to in this documentation, and you must ensure that your keyboard enables you to generate accurate EBCDIC hex codes. This is particularly true on keyboards that have been adapted to show local or national symbols. You should verify that

$ is mapped to x'5B'
# is mapped to x'7B'
@ is mapped to x'7C'

If you have any questions about whether your keyboard is properly mapped, contact your system administrator.

Preconfigured PFKeys

Many commands are preconfigured to specific keys or key combinations. This is particularly true with regard to numbered PF keys, or pairs of numbered PFKeys. For example, the END command is preconfigured to, and indicated as, **PF03/PF15**. To execute the END command, press either the **PF03** key or the **PF15** key.

Instructions to enter commands may include

- only the name of the command, such as, enter the END command
- only the PF keys, such as, press **PF03/PF15**
- or both, such as, press **PF03/PF15,** or enter the END command

Command Lines and Option Fields

Most screens contain a command line, which is primarily used to identify a single field where commands, or options, or both, are to be entered. These fields are usually designated COMMAND, but they are occasionally identified as COMMAND / OPT or COMMAND / OPTION.

Option field headings appear in many screens. These headings sometimes appear in the screen examples as OPTION, or OPT, or O.

Names of Commands, Fields, Files, Functions, Jobs, Libraries, Members, Missions, Options, Parameters, Reports, Subparameters, and Users

The names of commands, fields, functions, jobs, libraries, members, missions, options, parameters, reports, subparameters, users, and most files, are shown in standard UPPERCASE font.

User Entries

In situations where you are instructed to enter characters using the keyboard, the specific characters to be entered are shown in this **UPPERCASE BOLD** text, for example, type **EXITNAME.**

Syntax statements

In syntax, the following additional conventions apply:

- A vertical bar (|) separating items indicates that you must choose one item. In the following example, you would choose *a*, *b*, or *c*:

  ```
  a | b| c
  ```

- An ellipsis (. . .) indicates that you can repeat the preceding item or items as many times as necessary.

- Square brackets ([ ]) around an item indicate that the item is optional. If square brackets ([ ]) are around a group of items, this indicates that the item is optional, and you may choose to implement any single item in the group. Square brackets can open ([ ) and close ( ]) on the same line of text, or may begin on one line of text and end, with the choices being stacked, one or more lines later.

- Braces ({ }) around a group of items indicates that the item is mandatory, and you must choose to implement a single item in the group. Braces can open ({) and close (}) on the same line of text, or may begin on one line of text and end, with the choices being stacked, one or more lines later.

Screen Characters

`All syntax, operating system terms, and literal examples are presented in this typeface.` This includes JCL calls, code examples, control statements, and system messages. Examples of this are:

- calls, such as

```
CALL 'CBLTDLI'
```

- code examples, such as

```
FOR TABLE owner.name USE option, . . . ;
```

- control statements, such as

```
//PRDSYSIN DD * USERLOAD PRD(2) PRINT
```

- system messages, both stand-alone, such as `You are not logged on to database database_name`, and those embedded in text, such as the message `You are not logged on to database database_name`, are displayed on the screen.

Variables

Variables are identified with *italic* text. Examples of this are:

- In syntax or message text, such as
  `Specify database` *database_name*
- In regular text, such as
  replace database *database_name1* with database *database_name2* for the current session
- In a version number, such as
  EXTENDED BUFFER MANAGER for IMS 4.1.*xx*

Special elements

This book includes special elements called *notes* and *warnings*:

NOTE

Notes provide additional information about the current subject.

WARNING

Warnings alert you to situations that can cause problems, such as loss of data, if you do not follow instructions carefully.

Related Publications

CONTROL-D Getting Started Guide

Introduction to CONTROL-D concepts and facilities in the framework of a hands-on demonstration.

CONTROL-D Online Viewing Guide

Tutorial guide that demonstrates the features of the Online Viewing facility.

CONTROL-D Implementation Guide

Practical guide for determining implementation objectives, and for planning and performing the implementation of CONTROL-D.

Implementing AFP in the CONTROL-D Environment

Guide to the efficient utilization of the built-in AFP support features of CONTROL-D.

INCONTROL for z/OS Administrator Guide

Information for system administrators about customizing and maintaining INCONTROL™ products.

INCONTROL for z/OS Installation Guide

Step-by-step guide to installing INCONTROL products using the INCONTROL™ Installation and Customization Engine (ICE) application.

INCONTROL for z/OS Messages Manual

Comprehensive listing and explanation of all INCONTROL and IOA messages and codes.

INCONTROL for z/OS Security Guide

Step-by-step guide to implementing security in INCONTROL products.

INCONTROL for z/OS Utilities Guide

Describes utilities designed to perform specific administrative tasks that are available to INCONTROL products.

Chapter 1

Overview

This chapter includes the following topics:

Introduction

This publication helps you convert from CA-DISPATCH to CONTROL-D software.

The CA-DISPATCH to CONTROL-D conversion tool creates CONTROL-D components based on information extracted from the CA-DISPATCH database.

This conversion tool supports CONTROL-D version 6.1.00 and later. IOA and CONTROL-D environments must be installed before starting the conversion process.

Conversion Steps

The steps for converting CA-DISPATCH to CONTROL-D are described in Chapter 2, "Conversion Steps."

Each of these steps can be implemented separately according to the needs of the report distribution environment. For example, CA-DISPATCH archive indexes can be converted without performing other parts of the conversion.

Naming Conventions

CA-DISPATCH to CONTROL-D conversion members are located in the IOA SAMPLE library. Nearly all the members associated with this conversion tool have names beginning with the characters CAD.

- CADC*xxxx*—CA-DISPATCH CULPRIT report definitions
- CADDEF*xx*—Members containing default settings and definitions
- CADJ*xxxx*—Members containing conversion jobs
- CADL*xxxx*—CA-DISPATCH report layouts
- CADMS*xxxx*—Message member
- CADS*xxxx*—Source programs of the conversion jobs
- CADUSER—Sample of the user conversion table

Creating the CONTROL-D Recipient Tree

Building the CONTROL-D Recipient Tree

The CA-DISPATCH to CONTROL-D conversion tool provides two methods of building the CONTROL-D Recipient Tree.

Table 1 Methods for Building the CONTROL-D Recipient Tree

Method	Description
Method A	This method creates a 3-level CONTROL-D Recipient Tree. The first level, level 10, contains one user—CTDTREE. The second level, level 20, contains all the CA-DISPATCH MAILDROPs and their Addresses. The third level, level 30, contains all the CA-DISPATCH recipients. This method is recommended when the MAILDROP names differ from the recipient names and there are many recipients for each MAILDROP.
Method B	This method creates a 2-level CONTROL-D Recipient Tree. The first level, level 10 contains one user—CTDTREE. The second level, level 20, contains all the CA-DISPATCH MAILDROPs. The CA-DISPATCH recipients who belong to each MAILDROP are defined as synonyms of that level 20 MAILDROP recipient. Use this method if the MAILDROP names are the same as the recipient names and all 8 characters of the MAILDROP are used. This method is the recommended method because it creates a tree that enables CONTROL-D to function similarly to CA-DISPATCH.

Required Reports

The Recipient Tree conversion requires two standard CA-DISPATCH reports:

- Report Recipient File Listing (known as DSCULP02).
- Report Distribution Maildrop Listing - Defaults (known as DSCULP04).

These reports are generated on separate sequential files.

These two files are used as input to the tree building utility, CTDBLDTR. This program can build the CONTROL-D Recipient Tree from any DSCULP02 report. Specially tailored jobs to convert CA-DISPATCH reports using this program are contained in the IOA SAMPLE library, and described in Chapter 3, “Building a

CONTROL-D Recipient Tree." This utility can also be used also to add users to the CONTROL-D Recipient Tree from any other DSCULP02 report. Instructions on how to use the CTDBLDTR utility are also contained in the IOA SAMPLE library, and described in 3.

Program CADSTR04 is used to prepare the DSCULP04 report for processing by program CTDBLDTR.

Two jobs are supplied in the conversion tool source library:

- Job CADJTRE1 is the JCL for Recipient Tree creation Method A. Parameter members CADDEFT1, CADDEFT2 and CADDEFT3 are additional input for program CTDBLDTR.
- Job CADJTRE2 is the JCL for Recipient Tree creation Method B. Parameter members CADDEF4, CADDEFT5 and CADDEFT6 are additional input for program CTDBLDTR.

Creating Decollation Mission Definitions

Conversion program CTDSDECM creates CONTROL-D generic decollation mission definitions from CA-DISPATCH information. A report similar to the CA-DISPATCH Selection By Recipients Listing is used as input. These generic decollation missions enable CONTROL-D to decollate reports to the same recipients that received those Product Namee reports using CA-DISPATCH.

Printing characteristics are not processed by this conversion program because CONTROL-D automatically extracts all printing characteristics from the JES SPOOL. Therefore, the printing characteristics from the job's JCL are used.

Reports are converted as follows:

1. Using CADCDECM
2. Using CADJDECM

Using CADCDECM

Member CADCDECM contains a CULPRIT report definition that produces a list of all reports defined in the CA-DISPATCH database. This report contains the relevant information for creating CONTROL-D generic decollation definitions. The report is written to a disk file for later use by job CADJDECM. This report, used by the conversion program, is not the standard DSCULP13 report - Selection By Recipients Listing, because report DSCULP13 is not provided in all CA-DISPATCH releases. Therefore, a special CULPRIT report is supplied in this member.

Using CADJDECM

The first step of job CADJDECM defines all the files needed by the conversion program.

The second step sorts the output file created by job CADCDECM and places it on a new file.

Decollation mission definitions can be built by username or jobname. Sort the output file by REPORT NAME or JOBNAME according to whether you are building decollation mission definitions by username or jobname. The sort also eliminates blank records.

The third step in job CADJDECM creates the decollation mission definitions as follows:

- Job CADJDECM reads the sorted output file created by CADCDECM and produces members in a designated CONTROL-D REPORTS library. Each member is a CONTROL-D decollation mission. The name of each member is assigned the CA-DISPATCH report name (meaning, each CA-DISPATCH report has a corresponding CONTROL-D decollation definition member).

- Job CADJDECM activates program CTDSDECM to build the new CONTROL-D decollation mission definitions. The source code of this program resides in the SOURCE library and can be locally tailored. Program CTDSMEM is called to perform all the PDS operations while creating the new decollation mission definition members. This program receives the PARM parameter set to JOB or USER for building decollation missions based on jobname or username, accordingly.

Creating Printing Parameters

Using CADCOUTP

Member CADCOUTP contains a CULPRIT report definition that produces a list of all the recipients and their related reports, as defined in the CA-DISPATCH database. This report contains the recipient name, the recipient's reports, and the printing parameters used for each report. This information is extracted from CA-DISPATCH Type 3 LOAD records.

The report is written to a disk file for later use by jobs CADJOUTP and CADJAPAP. This report is used by the conversion programs. It is not a standard CA-DISPATCH report because none of the standard CA-DISPATCH reports contains all the printing characteristics information assigned for each report. Therefore, a special CULPRIT report is supplied in this member.

This report does not contain any JCL. You must add the correct JCL. Make sure that the output is routed to a disk file and properly referenced by DD statement SYS018 (as specified in the CULPRIT report). Specify the file destination only in the JCL and not in the CULPRIT options because the ASA code is used during the conversion process.

Using CADCDJDE

Member CADCDJDE contains a CULPRIT report definition that produces a list of all the recipients and their related reports, as defined in the CA-DISPATCH database. This report contains the recipient name, the recipient's reports, and all the DJDE parameters used for each report. This information is extracted from CA-DISPATCH LOAD records of Type 7, 8, 9, and A.

This report is written to a disk file for later use by job CADJDJDE. This report is used by the conversion program. It is not a standard CA-DISPATCH report because none of the standard CA-DISPATCH reports contains all the DJDE information assigned for each report. Therefore, a special CULPRIT report is supplied in this member.

This report does not contain any JCL. You must add the correct JCL. Make sure that the output is routed to a disk file, properly referenced by DD statement SYS018 (as specified in the CULPRIT report). Specify the file destination only in the JCL and not in the CULPRIT options because the ASA code is used during the conversion process.

Using CADJOUTP

Job CADJOUTP activates program CADSOUTP to build the new CONTROL-D OUTPARMS members. The source code of this program resides in the SOURCE library and can be locally tailored.

The job consists of four steps:

1. Defines all the files needed by the conversion program.

2. Updates the output created by job CADCOUTP and places the recipient and report names in each of the records. This enables the reports to be sorted by recipient name or report name.

3. Sorts the output from the previous step by USER NAME (in ascending order) and REPORT NAME (in ascending order), while keeping the original order of the records for each USER/REPORT NAME combination. This sort also eliminates blank records.

4. Creates the OUTPARMS parameter members in the CONTROL-D OUTPARMS library.

The CA-DISPATCH report contains CLASS DEFAULT and SPECIFIC CLASS fields. To support these fields, the last step uses an input parameter, CLASS DEFAULT, assigned using DD statement INPUT. If this input parameter contains a value and the SPECIFIC CLASS field in the CA-DISPATCH report does not, the value specified for input parameter CLASS DEFAULT is assigned to parameter CLASS. If input parameter CLASS DEFAULT is not assigned a value, then parameter CLASS is not added to the OUTPARMS member.

The OUTPARMS library supports JOBNAME in addition to USERNAME and REPORT NAME.

Using CADJAPAP

Job CADJAPAP activates program CADSAPAP to build the new CONTROL-D APAPARM members. The source code of this program resides in the SOURCE library and can be locally tailored.

Job CADJAPAP consists of two steps:

1. Defines all the files needed by the conversion program.

2. Creates the APA parameter members in the CONTROL-D APAPARM library.

Using CADJDJDE

Job CADJDJDE activates the CADSDJDE program to build the new CONTROL-D DJDEPARM members. The source code of this program resides in the SOURCE library, and can be locally tailored.

Job CADJDJDE consists of four steps:

1. Defines all the files needed by the conversion program.

2. Runs program CTDSCULU, which updates the output created by job CADCDJDE and places the recipient and report names in each of the records. This enables the reports to be sorted by recipient name and report name.

3. Sorts the output from the previous step by USER NAME (in ascending order) and REPORT NAME (in ascending order), while keeping the original order of the records for each USER/REPORT NAME combination. The sort also eliminates the blank records.

4. Creates the DJDE parameter members in the CONTROL-D DJDEPARM library.

Using CTDDJDE and CTDAPA

The CTDDJDE and CTDAPA routines that reside in the IOA SAMPEXIT library, support JOBNAME by default. Because the conversion programs build members in the APAPARM and DJDEPARM libraries by USERNAME, the CTDDJDE and CTDAPA routines support USERNAME. To make these routines work by USERNAME, activate optional wish WD2754 in the IOA MAC library.

Using CTDX003

After Using CTDDJDE and CTDAPA, activate the changes by re-assembling and link-editing User Exit CTDX003 into the IOA LOAD library. As of version 6.0.00 the exit is contained in the IOA SAMPEXIT library, and in versions 5.x.x, in the IOA SECUDATA library.

If CTDOUT is changed by this conversion, the user exit must be tailored and marked with OUTPARM set to USER (instead of being set to JOB or blanks). Otherwise, the needed members will not be found in the OUTPARMS library when printing a report.

NOTE

Program CADSMEM is used in the other conversion programs to perform all the PDS operations while creating the members in each new library.

Creating the CONTROL-D History User File

This part of the conversion process creates the History User file in the CONTROL-D environment to facilitate access to reports archived by CA-DISPATCH. This enables reports created by CA-DISPATCH to be restored in the CONTROL-D environment from the original tapes backed up by CA-DISPATCH.

Using CADCARC1 and CADCARC2 Members

Members CADCARC1 and CADCARC2 contain a CULPRIT report definition that produces a list of all reports defined in the CA-DISPATCH database and a list of all archived reports. These lists contain the relevant information for creating the CONTROL-D History User file. These lists should be written to a disk file for later use by job CADJARC1.

Before submitting the job, make sure that there is enough space specified for the new CONTROL-D REPORTS file (allocated in this step under ddname DAREPMIS).

Using Job CTDJARC1

Job CADJARC1 is an 8-step job that creates a sequential file for job CTDJARC2. Job CTDJARC2 builds the CONTROL-D History User file.

1. Defines all the files used by the conversion program using CADJARC1.

2. Sorts the CULPRIT report list of the definitions and remove unneeded records.

3. Sorts the table of user names. A sample user name table is supplied in member CADUSER.

4. Converts user names in the list of report definitions according to the user table in member CADUSER.

5. Removes duplicate records from the list of report definitions.

6. Reformats the CULPRIT report list of archived reports using the CADCARC1 report format.

7. Merges both reports into one report to be used by job CTDJARC2.

8. Propagates user names for all records in the resulting report to be used by job CTDJARC2.

Using Job CTDJARC2

Job CADJARC2 creates records in the CONTROL-D History User file.

- Defines all the files needed by the conversion program using CADJARC2.

- Creates records in the CONTROL-D History User file using CADJARC2 based on input from job CTDJARC1.

This job adds records to an existing History User file. If this job is rerun, you must reformat the History User file to prevent the addition of duplicate records. Use job CTDUFDBF from the CTD JCL library to reformat the History User file.

User Exit CTDX004

Adjust User Exit CTDX004 if archived reports are converted. Exit CTDX004 receives control during the restore request and starts a process for restoring reports from CA-DISPATCH tapes. As of version 6.0.00, a sample of User Exit CTDX004 is supplied in the IOA SAMPEXIT library. In versions 5.x.x, the exit is located in the IOA SECUDATA library.

Exit CTDX004 submits a job to locate the corresponding report on the tape, writes this report directly to a CDAM file, and creates new user and sysdata records in the Active User file.

The programs invoked by this job are located in the IOA LOAD library.

Using CADSKL Skeleton Job

CADSKL is a skeleton for building a job to restore reports from CA-DISPATCH tapes. This skeleton is located in the CONTROL-D SKL library.

Special Considerations

1. The USER NAME in the CULPRIT report has the same characteristics as the USER NAME in CA-DISPATCH files (16 characters maximum, blanks allowed). To adjust the USER NAME to the CONTROL-D environment, the name is truncated to 8 characters and blanks are replaced by underscores ("_"). This process is applied to the Recipient Tree conversion, but the full USER NAME is set to one of the synonyms in the tree.

2. The function of the PAGE FLAG in CA-SAR is implemented by AND / OR logic in the WHEN statement, and by the following CONTROL-D parameters:

   CONTID (Y / N).
   REFER TO NEXT PAGE (Y / N).
   PRINT (Y / N).

   Special user name NULL must be defined in the Recipient Tree to support the REFER TO NEXT PAGE option.

3. The ARCHIVE option in CA-DISPATCH is converted to the BACKUP option in CONTROL-D. When N is specified in the CA-DISPATCH report definition, no backup mission name is inserted in the report decollation definition. Otherwise, the default BACKUP mission name, specified as an external parameter in member CADDEFDM, is used in the DO BACKUP statement.

4. The DO NAME statement in each CONTROL-D report decollation definition is set to the CA-DISPATCH report name.

5. In each ON CLASS statement, parameter EXTWTR is set to the CA-DISPATCH report name. The report decollation is then made to this EXTWTR.

6. If a string is specified for TEST purposes in the CA-DISPATCH report, the same test is performed in CONTROL-D. In addition, the conversion enables you to open a window for a string search. For more information, see "#LINES RANGE".

7. There is a maximum default value for the number of cards for each member built by this conversion. This number is specified in routine CADSDECM in the #CARDS and AREALEN constants. If necessary, this value can be locally tailored.

8. Each member created in a DJDEPARM, APAPARM, or OUTPARMS library is assigned a member name according to the following criteria:

   - If the report name contains only three characters, the site is using a Universal Writer name. The member name is set to $xxx, where xxx is the report name. Because only one occurrence of the member can exist, the first occurrence found in the CULPRIT report is used. Any later occurrence of the same report name in the CULPRIT report is ignored.

- If the report name starts with @ or #, the site is using a Unique report name. The member name is set to @xxxxxxx or #xxxxxxx, where xxxxxxx is the report name. Because only one occurrence of the member can exist, the first occurrence found in the CULPRIT report is used. Any later occurrence of the same report name in the CULPRIT report is ignored.

- For all other report names, the member name is set to the USERNAME, subject to the above restrictions (meaning, the name is truncated to 8 characters, embedded blanks are replaced by underscores, and trailing blanks are eliminated).

9. DJDE parameters ITEXT, OTEXT and RTEXT can contain a maximum of 82 characters in field SC. Because the DJDEPARM, APAPARM and OUTPARMS libraries have a LRECL of 80, the text in field SC is truncated to the maximum size line each member can contain.

10. If the same report is assigned to a user more than once, only the first occurrence of the report record is used. Duplications are ignored. However, if different occurrences of the same report name assigned to a specific user contain the CHARS option, it is added to the OUTPARMS member only if the first record also contained the CHARS option.

11. The contents of each member is constructed as follows:

    - For a Universal Writer report or Unique report, the corresponding member contains

      —for DJDE parameters:

```
          +++*
            DJDE JDE=ABCDEF,JDL=GHIJKL,;
            DJDE DEPT='EXAMPLE OF DEPT FIELD',END;
```

      —for OUTPARMS parameters:

```
          +++*
            CLASS=X
            CHARS=(CHR1,CHR2)
```

    - Any other member contains

      —for DJDE parameters:

```
+++report-name-1
  DJDE JDE=ABCDEF,JDL=GHIJKL,;
  DJDE DEPT='EXAMPLE OF DEPT FIELD',END;
+++report-name-2
  DJDE JDE=XXXXXX,JDL=YYYYYY,;
  DJDE FONTS=((FONTS1,12LPI),(FONTS2,11DOTS)),END;
     .
     .
```

— for OUTPARMS parameters:

```
+++report-name-1
CLASS=X
CHARS=(CHR1,CHR2)
+++report-name-2
CLASS=Y
OPTCD=J
FCB=ZZZ
```

12. Carefully check the CADLDJDE DSECT, describing the DJDE CA-DISPATCH record (Types 7, 8, 9, and A) before submitting job CADJDJDE, because the local tests performed for these record types do not contain complete information. Therefore, the field offsets can be incorrect. If the CADLDJDE DSECT is changed, programs CTDSDJDE and CTDSCULU must be re-assembled and link-edited into the conversion tool LOAD library.

    Differences between the supplied CULPRIT output reports and the resulting reports at each site can be due to the fact that different versions of CA-DISPATCH produce different results. This also applies to the CADMOUTP DSECT, which maps the CA-DISPATCH Type 3 LOAD records. If this DSECT is changed, the CTDSOUTP, CTDSAPAP and CTDSCULU programs must be re-assembled.

    When changing the offsets within each DSECT, take special care with the SORT FIELDS of each conversion job, because the SORT key is based on a combination of USERNAME and REPORT NAME. An invalid sort can produce incorrect results and error messages while building the OUTPARMS, APAPARM and DJDEPARM libraries.

13. This conversion was designed with the assumption that the JOBNAME field is seldom used in the CA-DISPATCH database. Therefore, all members in the various libraries are named according to USERNAME. If the JOBNAME field is used at your site for every report entry, then:

    - The conversion routines can locally be tailored to support JOBNAME as the member name.

    - Change the SORT FIELDS in conversion jobs CADJOUTP and CADJDJDE to support JOBNAME as the primary sort key and REPORT NAME as the secondary sort key.

- CTDAPA and CTDDJDE can remain as supplied in the installation tape. However, CTDAPA will not then support report name masking.

- CTDOUT must be referenced by User Exit CTDX003 using parameter OUTPARM set to JOB.

14. Set the default parameters in member CADDEFDM before job CADJARC2 is started.

    Do not change the expressions PRODUCT=CA-DISPATCH and CATEGORY=CA-CONVERTED. User Exit CTDX004 inserts the CATEGORY field in the corresponding field in the USER records for additional analysis by User Exit CTDX004. Based on this parameter, Exit CTDX004 determines whether to use a special restore process.

15. Handling the Universal Report Character.

    When using the CA-DISPATCH Universal Report option, all the report decollation missions created for this type of report can be omitted from the new CONTROL-D REPORTS library. It is easy to identify these “Universal” report decollation missions because they all have a 3-character report name.

    These reports are decollated using one generic decollation mission. This special generic decollation mission is supplied in member CADUNIVR in the conversion tool SOURCE library. Copy member CADUNIVR manually to the new CONTROL-D REPORTS library.

Chapter 2

Conversion Steps

This chapter includes the following topics:

Overview

The conversion process consists of the following steps, which can be implemented separately according to the needs of the report distribution environment.

1 Create the CONTROL-D Recipient Tree from CA-DISPATCH recipient reports. The CONTROL-D Recipient Tree is a very important element of CONTROL-D. It is used by almost all CONTROL-D processes. Therefore, the Recipient Tree should include all CONTROL-D recipients before you begin testing CONTROL-D functions.

A Create CA-DISPATCH Report DSCULP02

B Create CA-DISPATCH Report DSCULP04

C Tailor and Run Job CADJTRE1 or CADJTRE2

2 Create CONTROL-D decollation mission definitions from CA-DISPATCH information, as follows:

A Tailor and Run Job CADCDECM

B Check and / or Modify Options Specified in Member CADDEFDM

C Tailor and Run Job CADJASML

D Tailor and Run Job CADJDECM

E Copy and Tailor Member CADUNIVR

3 Create printing parameters for the various recipients, using report relations that exist in CA-DISPATCH. The relationships and their parameters are translated into DJDE parameters (if they exist in the CA-DISPATCH database) and to OUTPARMS and APA parameters.

A Tailor and Run Job CADCOUTP

B Tailor and Run Job CADCDJDE

C Tailor and Run Job CADJOUTP

D Tailor and Run Job CADJAPAP

E Tailor and Run Job CADJDJDE

F Activate Optional Wishes WD1643 and WD2754

4 Create the CONTROL-D History User file from CA-DISPATCH information, to enable access to reports archived by CA-DISPATCH.

A Tailor and Run Jobs CADCARC1 and CADCARC2

B Tailor and Run Jobs CADJARC1 and CADJARC2

C Tailor and Recompile User Exit CTDX004

D Tailor Skeleton CADSKL in the CONTROL-D SKL Library

5 Test the Conversion

Step 1 Create CONTROL-D Recipient Tree

Step 1.A Create CA-DISPATCH Report DSCULP02

1. Create the CA-DISPATCH report: Report Recipient File Listing. Use standard CA-DISPATCH member DSCULP02.

   The DD statement for the report output must reference a dataset definition and not a sysout. Write the CULPRIT report to a sysout but ensure that the Report Recipient File Listing is routed to a file and not a sysout.

2. Name the output file CADI.REPORT02. Otherwise, you must change the name in jobs CADJTRE1 and CADJTRE2. The output file must have the following characteristics: physical sequential, RECFM FBA, LRECL 250.

Step 1.B Create CA-DISPATCH Report DSCULP04

1. Create the CA-DISPATCH report: Report Distribution Maildrop Listing – Defaults. Use standard CA-DISPATCH member DSCULP04.

   The DD statement for the report output must reference a dataset definition and not a sysout. Write the CULPRIT report to a sysout but make sure that the Report Distribution Maildrop Listing – Defaults is routed to a file and not a sysout.

2. Name the output file CADI.REPORT04. Otherwise, you must change the name in jobs CADJTRE1 and CADJTRE2 jobs. The output file must have the following characteristics: physical sequential, RECFM FBA, LRECL 250.

Step 1.C Tailor and Run Job CADJTRE1 or CADJTRE2

1. Tailor member CADJTRE1 or member CADJTRE2 to create a Recipient Tree using Method A or Method B. For more information about these two methods, see Chapter 1, “Required Reports” in 1.

2. Submit the job for execution and check the sysout for error messages. All job steps must end with a condition code of 0, except for step 1, which can terminate with a condition code of 12.

Step 2 Create Decollation Mission Definitions

Step 2.A Tailor and Run Job CADCDECM

1. Tailor member CADCDECM in the IOA SAMPLE library. This member is a CULPRIT report definition.

2. Create the JCL for running this report. The DD statement for the report output must reference a dataset definition and not a sysout. Write the CULPRIT report to a sysout but make sure that the output of job CADCDECM is routed to a file and not a sysout.

   The following table describes the input and output for this step:

Table 2 Job CADCDECM Input and Output

Data	Description
Input	CA-DISPATCH Database.
Output	A sequential file containing the report Name the output file CTD.CADI.REPORT. Otherwise, you must change the name in job CADJDECM. The output file must have the following characteristics: physical sequential, RECFM FBA, LRECL 279

3. Submit the job for execution and check the sysout error messages. The job must end with a condition code of 0.

Step 2.B Check and/or Modify Options Specified in Member CADDEFDM

Tailor the CONTROL-D options specified in member CADDEFDM of the IOA SAMPLE library.

Step 2.C Tailor and Run Job CADJASML

1. Use member ASMLINK to assemble and link-edit the conversion programs.

2. Tailor the JCL of this member according to your naming conventions.

3. Submit the job for execution and check the sysout for condition code and error messages. The link-edit steps of this job must end with a condition code of 0.

Step 2.D Tailor and Run Job CADJDECM

This job creates the CONTROL-D REPORTS library for report decollation mission definitions.

The following table describes the input and output for this step:

Table 3 Job CADJDECM Input and Output

Data	Description
Input	CTD.CADI.REPORT file, created by CADCDECM (as described in Step 2.A above). Member CADDEFDM in the IOA SAMPLE library. This member contains external parameters for the conversion. These parameters are used as defaults for the report decollation definitions.
Output	A PDS library containing CONTROL-D generic report decollation mission definitions. The default file name is CTD.REPORTS. The output file has the following characteristics: partitioned dataset, logical record length 80, blocksize 3120.

1. Tailor member CADJDECM in the IOA SAMPLE library.

NOTE

Ensure that the SPACE parameter specified for the DAREPMIS file contains enough directory blocks and the primary allocation value is large enough.

The job can run for quite some time, depending on the number of reports in the CULPRIT report file.

2. Submit the job for execution and check the sysout for error messages. The job must end with a condition code of 0.

Step 2.E Copy and Tailor Member CADUNIVR

1. Copy member CADUNIVR from the conversion tool source library into the CONTROL-D REPORTS library, which was created by job CADJDECM. Fill in the OWNER ID, GENERIC ON CLASS, BACKUP mission name, and the CADUNIVR character used by your site in the EXTWTR field.

2. Save the member.

   This generic decollation mission is used to decollate all reports that were processed in CA-DISPATCH using the Universal Report Writer.

Step 3 Create Printing Parameters

Step 3.A Tailor and Run Job CADCOUTP

1. Tailor member CADCOUTP in the IOA SAMPLE library. This member is a CULPRIT report definition. Ensure that the output is routed to a file and not to a sysout. DD statement SYS018 references the output file. The report does not contain any JCL. You must add the correct JCL.

   Do not change the CULPRIT definition itself. Otherwise, the results are unpredictable. Specify the file destination only in the JCL and not in the CULPRIT options because the ASA code is used during the conversion process.

   The following table describes the input and output for this step:

Table 4 Job CADCOUTP Input and Output

Data	Description
Input	CA-DISPATCH database
Output	A sequential file containing the report Name the output file CTD.CADI.OUTPUT. Otherwise, you must change the name in jobs CADJOUTP and CADJAPAP. The output file must have the following characteristics: physical sequential, record format FBA, logical record length 279.

2. Submit the job for execution and check the sysout for error messages. The job must end with a condition code of 0.

Step 3.B Tailor and Run Job CADCDJDE

1. Tailor member CADCDJDE in the IOA SAMPLE library. This member is a CULPRIT report definition. Do not change the CULPRIT definition. Otherwise, results are unpredictable. Ensure that the output is routed to a file and not to a sysout. DD statement SYS018 references the output file.

   The report does not contain any JCL. You must add the correct JCL. Specify the file destination only in the JCL and not in the CULPRIT options because the ASA code is used during the conversion process.

   The following table describes the input and output for this step:

Table 5 Job CADCDJDE Input and Output

Data	Description
Input	CA-DISPATCH database.
Output	A sequential file containing the report Name the output file CTD.CADI.DJDE. Otherwise, you must change the name in job CADJDJDE. The output file must have the following characteristics: physical sequential, record format FBA, logical record length 1300.

2. Submit the job for execution and check the sysout for error messages. The job must end with a condition code of 0.

Step 3.C Tailor and Run Job CADJOUTP

This step creates the CONTROL-D OUTPARMS library whose members contain the regular printing characteristics for each report.

The following table describes the input and output for this step:

Table 6 Job CADJOUTP Input and Output

Data	Description
Input	CTD.CADI.OUTPUT file created by the job in member CADCOUTP.
Output	A PDS library, containing CONTROL-D OUTPARMS members. Default file name: CTD.OUTPARMS. The characteristics of the file are: Partitioned dataset, logical record length 80, blocksize 3120.

1. Tailor member CADJOUTP.

NOTE

Ensure that the SPACE parameter specified for the DAOUTPUT file contains enough directory blocks and the primary allocation value is large enough.

The job can run for quite some time, depending on the number of reports in the CULPRIT report file.

2. Submit the job for execution and check the sysout for error messages. The job must end with a condition code of 0.

Step 3.D Tailor and Run Job CADJAPAP

This step creates the CONTROL-D APAPARM library whose members contain the APA control statements for each report.

The following table describes the input and output for this step:

Table 7 Job CADJAPAP Input and Output

Data	Description
Input	CTD.CADI.OUTPUT file created by the job in member CADCOUTP
Output	A PDS library containing CONTROL-D APAPARM members Default file name: CTD.APAPARM. The output file must have the following characteristics: Partitioned dataset, logical record length 80, blocksize 3120.

1. Tailor member CADJAPAP according to your local conventions:

NOTE

Ensure that the SPACE parameter specified for the DAAPA file contains enough directory blocks and the primary allocation value is large enough.

The job can run for quite some time, depending on the number of reports in the CULPRIT report file.

2. Submit the job for execution and check the sysout for error messages. The job must end with a condition code of 0.

Step 3.E Tailor and Run Job CADJDJDE

This step creates the CONTROL-D DJDEPARM library, whose members contain the DJDE control statements for each report.

The following table describes the input and output for this step:

Table 8 Job CADJDJDE Input and Output

Data	Description
Input	CTD.CADI.DJDE file created by the job in member CADCDJDE
Output	A PDS library containing CONTROL-D DJDEPARM members Default file name: CTD.DJDEPARM. The output file must have the following characteristics: Partitioned dataset, logical record length 80, blocksize 3120.

1. Tailor the member CADJDJDE according to your local conventions.

NOTE

Ensure that the SPACE parameter specified for the DADJDE file contains enough directory blocks and the primary allocation value is large enough.

The job can run for quite some time, depending on the number of reports in the CULPRIT report file.

2. Submit the job for execution and check the sysout for error messages. The job must end with a condition code of 0.

Step 3.F Activate Optional Wishes WD1643 and WD2754

1. Activate optional wish WD1643.

   By default, CONTROL-D counts blank lines in decollation missions. When this wish is activated, CONTROL-D will no longer count blank lines in decollation missions. This is similar to CA-DISPATCH, which also does not count blank lines. This wish functions only for decollation missions with the group name SKIPBLANK.

2. Activate optional wish WD2754.

   Re-assemble user exits CTDX003 and CTDX014 from the IOA SAMPEXIT library, as of version 6.0.00. In versions 5.x.x, these exits are located in the IOA SECUDATA library.

   For more information about this wish, see member CTDDFLTS in the IOA DOC library.

Step 4 Create CONTROL-D History File

Step 4.A Tailor and Run Jobs CADCARC1 and CADCARC2

1. Tailor members CADCARC1 and CADCARC2 in the IOA SAMPLE library. These members are CULPRIT report definitions. Adjust them according to your local conventions. Create the JCL for running this report. Ensure that the output is routed to a file and not to a sysout.

2. Write the CULPRIT report to a sysout. Only the DD statement specified for this report must contain a dataset definition and not a sysout. The reason for this is based on ASA code considerations in the conversion program.

   The following table describes the input and output for this step:

Table 9 Jobs CADCARC1 and CADCARC2 Input and Output

<table>
<tr><th>Data</th><th>Description</th></tr>
<tr><td>Input</td><td>CA-DISPATCH database</td></tr>
<tr><td>Output</td><td>Two sequential files containing the reports
Name the first output file (created by job CADCARC1) CTD.CADI.OUTREP. Otherwise, you must change the name in job CADJARC1.
The output file must have the following characteristics: physical sequential, record format FB, logical record length 80.
The file format is illustrated below:</td></tr>
<tr><td colspan="2"><pre>
----+----1----+----2----+----3----+----4----+----5----+
         REPORT   JOB   RECIPIENT
                                  BEAUCHAT TOM

        ADAREP   GPADA004   AIS ARCUS
        ADAREP   GPADA004   TECH   SUPPORT
</pre></td></tr>
<tr><td></td><td>Name the second output file (created by job CADCARC2) CTD.CADI.HISREP. Otherwise, you must change the name in job CADJARC1.
The output file must have the following characteristics: physical sequential, record format FA, logical record length 133.</td></tr>
</table>

3. Submit the job for execution and check the sysout for error messages. The job must end with a condition code of 0.

Step 4.B Tailor and Run Jobs CADJARC1 and CADJARC2

1. Tailor and run jobs CADJARC1 and CADJARC2. Check the sysouts for error messages. The link-edit steps of these jobs must end with a condition code of 0.
2. Run the CTDUFSR utility to resort the data portion of the History User file. A sample job can be found in member CTDUFSR in the CTD JCL library.

Step 4.C Tailor and Recompile User Exit CTDX004

Adjust user exit CTDX004. Use sample exit CTDX004L supplied in the IOA SAMPEXIT library, as of version 6.0.00. In versions 5.x.x, this exit is located in the IOA SECUDATA library.

Step 4.D Tailor Skeleton CADSKL in the CONTROL-D SKL Library

Tailor skeleton CADSKL in the CTD SKL library.

Step 5 Test the Conversion

Test the conversion.

Chapter 3

Building a CONTROL-D Recipient Tree

This chapter includes the following topics:

Overview

Use the CTDBLDTR utility to create or modify the CONTROL-D Recipient Tree. This utility uses input from two sources: a report (referenced by DD statement REPORT), and a set of instructions (referenced by DD statement SYSIN) specifying how the data in the report is used to create users in the Recipient Tree.

This utility produces a list (referenced by DD statement SYSPRINT) summarizing the structure (input supplied by the user in SYSIN), and the Recipient Tree (referenced by DD statement TREE). TREE is a member of a partitioned dataset. If TREE is an empty member, the utility creates the Recipient Tree. If TREE contains an existing Recipient Tree, the utility modifies it.

The utility scans each line of the REPORT input and processes it according to the specifications included in the SYSIN data.

For sample JCL programs to execute utility CTDBLDTR, see the CTDBLDDC and CTDBLDJB members in the IOA SAMPLE library.

Defining the Levels

The instruction syntax for building the Recipient Tree is as follows:

Figure 1 Instruction Syntax for Building the Recipient Tree

```
LEVEL=xx
USER -
POS=n
LENGTH=n
{    DEFAULT=ccc }
{       POS=n
LENGTH=n
{    DEFAULT=ccc } }
.
.
.
{       POS=n
LENGTH=n
{    DEFAULT=ccc } }
{ PARENT -
PLEVEL=xx
POS=n
LENGTH=n
{    DEFAULT=ccc }
{    TRANSLATE=tabledd } }
{ ADDRESS -
POS=n
LENGTH=n
{    DEFAULT=ccc } }
{ SYNONYM -
POS=n
LENGTH=n
{    DEFAULT=ccc } }
{ SYNONYM -
POS=n
LENGTH=n
{    DEFAULT=ccc } }
END
```

Each LEVEL command can contain four types of paragraphs:

Table 10 LEVEL Command Paragraphs

Paragraph	Description
USER	Instructions to construct the user name. Mandatory.
PARENT	Instructions to construct the parent name. Optional.
ADDRESS	Instructions to construct the address text. Optional.
SYNONYM	Instructions to construct synonyms. Optional. Can be used more than once to construct more than one synonym for each user.

Parameters define how to process the paragraph. These parameters are repeated for the same paragraph if the data to be constructed consists of data contained in more than one string in the report line.

The use of these parameters is as follows:

Table 11 LEVEL Command Parameters

Parameter	Description
POS	Starting character position of the string in the input report. A value of 0 (zero) indicates that the default value is used. Data from the report is not used for this parameter. The POS value is relative to the first print column of the report (that is, for the first column of the report, POS is set to 1) and does not include print control characters or variable record length values.
LEN	Length of the data extracted from the input report, starting from the character position specified in POS or from the default value if POS is set to 0.
DEFAULT	The default value. Optional. Any position in the field of length LEN from the report that is blank is replaced by the corresponding character from parameter DEFAULT.

The combined total length of the data constructed from all the repetitions of the parameters, for each paragraph, must not exceed the number of characters shown in the table below:

Table 12 Maximum Number of Characters for Paragraph Types

# Characters	Paragraph Type
8	USER
8	PARENT
52	ADDRESS
20	SYNONYM

The utility constructs the users based on these definitions and searches the Recipient Tree to see if the constructed user is already defined. If the user is not found, the utility adds the user. If the user is found, the utility updates the Recipient Tree. This utility is especially useful if the only changes required are the addition of synonyms.

Additional Considerations for the PARENT Paragraph

The PARENT paragraph has two additional parameters:

Table 13 PARENT Paragraph Parameters

Parameter	Description
PLEVEL	Mandatory. Level at which the parent is located.
TRANSLATE	Optional. DD name referencing a file containing the USER/PARENT correspondence.

If the parent of a user cannot be identified from the data on the report line, but can be determined from the user name, a file containing a table relating parent names to user names is supplied.

Each line in the external table is in the format USER=*usermask* PARENT=*parent*

In the *usermask* field, masking characters have the following meaning:

- *—Matches any number of consecutive characters.
- ?—Matches any one character.

Examples

- USER=ABC*D matches users ABC*123*D, ABC*12*D, and ABC*X*D.
- USER=ABC?D only matches user ABC*X*D from the above set.

TREE Construction Example

Suppose that the SYSIN file contains the following statements:

Figure 2 TREE Construction Example

```
LEVEL=20
USER -
POS=0
LENGTH=1
DEFAULT=L
POS=10
LENGTH=2
DEFAULT=03
PARENT -
PLEVEL=10
POS=0
LENGTH=6
DEFAULT=CDTREE
LEVEL=30
PARENT -
PLEVEL=20
POS=0
DEFAULT=L
LENGTH=1
POS=10
LENGTH=2
TRANSLATE=TABLE1
USER -
POS=0
LENGTH=1
DEFAULT=L
POS=1
LENGTH=4
DEFAULT=CKJ
SYNONYM -
POS=0
LENGTH=4
DEFAULT=USER
POS=1
LENGTH=4
SYNONYM -
POS=0
LENGTH=5
DEFAULT=SYN1-
POS=1
LENGTH=4
```

The Recipient Tree is constructed as follows:

For each line in REPORT file:

1. For a user at level 20 with the name L*xy*, where *xy* are the contents of columns 10 and 11 in the report line:

   - If column 10 is blank, the name is ‘L0*y*’
   - If column 11 is blank, the name is ‘L*x*3’
   - If both are blank, the name is ‘L03’

2. The parent of this user is at level 10 with the name CDTREE.

3. For a user at level 30 with the name L*abcd*, where *abcd* are the contents of columns 4 through 7 in the report line:

   Default CKJ is used in a manner similar to default 03 in item 1 above.

4. An attempt is made to determine a level 20 parent from the value ‘L’ plus the contents of columns 10 and 11. If this does not succeed, then the file referenced by DDNAME TABLE1 is scanned line by line until a match is found for the user name and the parent name are taken from the file.

   Two synonyms are created: ‘USER*abcd*’ and ‘SYN1-*abcd*’, where abcd represents the contents of columns 1 through 4 in the report line.

Appendix

A

Default Conversion Parameters

Default definition parameters for decollation missions are contained in member CADDEFDM in the IOA SAMPLE library. These parameters can be tailored according to the needs of your site.

Table 14 Member CADDEFDM Default Decollation Definition Parameters (part 1 of 2)

Parameter	Description
ON CLASS	Mandatory. Describes the classes on which this report can be located. Maximum length: 8 characters. Specify at least one class.
DEFAULT USER	Optional. Specifies a valid user name, defined in the CONTROL-D Recipient Tree, that gets the unidentified pages of a report. Maximum length: 8 characters.
DEFAULT COPIES	Optional. Defines the default number of copies to produce when printing the report. If not specified, the value 098 is taken from the DEFAULTS definitions. For more information, see the *CONTROL-D User Guide*. Parameter length: 3 characters. Leading zeroes must be used.
MAX COPIES	Optional. Defines the maximum number of copies. If not specified, the value of 098 is taken from the DEFAULTS definitions. For more information, see the *CONTROL-D User Guide*. Parameter length: 3 characters. Leading zeroes must be used.
CATEGORY	Mandatory. Defines a report decollation mission category name. By default, the category name is set to a JOBNAME if one exists in the CULPRIT report. If not, the category name is taken from this parameter. Maximum length: 20 characters.
OWNER	Mandatory. Defines the default USER ID to which reports are assigned. Maximum length: 8 characters.
PRINT BY FORM	Optional. Indicates whether the printing mission name is set to ■ the FORM name taken from the CA-DISPATCH CULPRIT report, ■ or the default printing mission name (set to STD). Valid values: Y (Yes) for the FORM name or N (No) for STD.

Table 14 Member CADDEFDM Default Decollation Definition Parameters (part 2 of 2)

Parameter	Description
BACKUP MISSION	Optional. Specifies the BACKUP MISSION name to be used if the Archive option is set to Y in the CULPRIT report. Maximum length: 8 characters.
MIGRATION MISSION	Optional. Specifies the MIGRATION MISSION name. Maximum length: 8 characters.
#LINES RANGE	Optional. Defines the "window" in which to search for a string within the page. For example, if "from line" is set to 001 and #LINES RANGE is set to 003 in the CA-DISPATCH report, then the string in CONTROL-D is searched from lines 001 through 003. The value 000 means: search only in the designated line. Parameter length: 3 characters. Use leading zeroes. Default: 000
RETRO	Optional. Retroactive scheduling. Specifies whether to schedule a report decollation mission if its original schedule date has passed. Default: "*", means do not schedule the mission.
MAXWAIT	Optional. Number of days to wait for report decollation completion. Specifies the number of "extra" days a decollation mission waits for execution in the Active Missions file. After the mission has waited the extra days, it is deleted. Default: 0 (days)
GENERIC	Do not modify GENERIC.
VERSION	Do not modify VERSION.
MONTHS	For future use. Do not modify MONTHS.
COPIES	Optional. Sets the default number of copies if this parameter is not specified in member INPARM. Default: 98.
DEFPRT	Default print mission name used in the DO PRINT statement in the decollation definition.
LINES	Optional. Default number of lines used as a "window" for a string search. Default: 000 (no window)

Default archive conversion parameters are contained in member CADDEFAR in the IOA SAMPLE library. Tailor these parameters according to the requirements of the site.

Table 15 Member CADDEFAR Default Archive Conversion Parameters (part 1 of 2) (part 1 of 2)

Parameter	Description
PRODUCT	Default CA-DISPATCH. Do not modify.
CATEGORY	Default CA-DISPATCH-CONVERTED. Do not modify.
CLASS	Optional.
COPY#	Optional.
FORM	Optional.
CHARS	Optional.

Table 15 Member CADDEFAR Default Archive Conversion Parameters (part 2 of 2) (part 2 of 2)

Parameter	Description
MODIFY	Optional.
DEFRETP	Default retention period. Default: 0110 (days)

Messages

CTDCAD01S **BAD RC=*rc* FROM PUTMEM FUNCTION. MEMBER – *memname***

Explanation: An error occurred during processing by CONTROL-M routine CTMMEM. The error probably occurred as a result of insufficient space in the CONTROL-D REPORTS library or the DJDEPARM, APAPARM, or OUTPARMS library.
Routine CADSMEM is used to perform all the required operations on PDS libraries and members.

System action: The job terminates.

User response: Determine which library member was being processed and take appropriate corrective action.

CTDCAD02E **INVALID INPUT PARM *param***

Explanation: The external input parameters list contains an invalid parameter. Valid options for input parameters are listed in the conversion routine.

User response: Determine which parameter is not valid and correct it.

CTDCAD03E **MISSING VALUE FOR PARM *param***

Explanation: The parameter listed in this message is mandatory.

User response: See the description of the missing or invalid input parameter elsewhere in this guide. Specify a valid value for the required parameter.

CTDCAD04E **MISSING OBLIGATORY PARAMETER**

Explanation: Member INPARM in the conversion library contains several mandatory parameters. At least one of them is missing.

User response: See the description of the missing parameters elsewhere in this guide. Specify a valid value for the parameters.

CTDCAD05E NO MORE SPACE FOR REPORT: *rpt*. PROCESSING NEXT REPORT

Explanation: Report definition member *rpt* contains more lines than specified in conversion routine CTDCAD01, constants #CARDS, and AREALEN.

System action: The member is processed only to the specified line limit. The remaining lines are skipped. Processing continues with the next report.

User response: Increase the value of the parameter, rerun job ASMLINK, and rerun CADIDMIS.

CTDCAD06E GETMAIN FOR AREA FAILED

Explanation: A memory acquisition MVS function failed. The value specified for JCL parameter REGION is not large enough.

User response: Increase the value of parameter REGION and rerun the failing job.

CTDCAD07E FREEMAIN OF AREA FAILED

Explanation: Allocated memory cannot be freed.

System action: The conversion routine terminates with a non-zero return code.

CTDCAD08E DD CARD *ddname* COULD NOT BE OPENED

Explanation: A required DD statement is probably missing from the JCL of job CADJDECM.

User response: Supply the missing DD statement and rerun the job.

CTDCAD09S ERROR PROCESSING DIRECTORY

Explanation: This WTO message is generated by routine CADSMEM, which handles PDS operations. The CONTROL-D REPORTS library reached its directory limit.

User response: Re-allocate the REPORTS file with a larger number of directory blocks and resubmit job CADIDMIS.

CTDCAD10S DEFAULT COPIES NUMBER IS GREATER THEN THE MAX COPIES NUMBER. 98 IS ASSUMED

Explanation: The default value of parameter DEFAULT COPIES in member CADDEFDM is greater then the value specified for parameter MAX COPIES.

User response: Specify compatible values for parameters DEFAULT COPIES and MAX COPIES.

CTDCAD11E APAPARM MEMBER *memname* IS OUT OF SPACE. PROCESSING NEXT REPORT

Explanation: Report definition member *memname* contains more lines than specified in the main conversion routine CTDSAPAP, constants #CARDS, AREALEN, #CARDSV, AREAVLEN, #CARDSQ, SIZEREG, SIZEV, SIZELV, SIZELQ, and AREAQLEN.

System action: The member is processed only to the specified limit. The rest of the information is skipped.

User response: Increase the value of the relevant parameters, rerun job ASMLINK, and rerun job CADJAPAP.

CTDCAD12E *internal-mem* LIST IS OUT OF SPACE

Explanation: While processing a Universal Writer report or a Unique report, the corresponding internal member list (either the Universal or Unique list) ran out of space.
The maximum number of entries for each list is specified in all three conversion programs (CADSOUTP, CADSAPAP, CADSDJDE). This list is used as an index for the existing Universal / Unique members in each printing parameters library.

System action: When reaching the specified limit of the list, processing stops, the files are closed and the job terminates.

User response: Enlarge the value specified in constant LISTVLEN (for Universal) or LISTQLEN (for Unique) in all three programs. Rerun job ASMLINK and rerun the failing job.

CTDCAD13E OUTPARMS MEMBER *memname* IS OUT OF SPACE. PROCESSING NEXT REPORT

Explanation: Report member definition *memname* contains more lines than specified in main conversion routine CADSOUTP, constants #CARDS, AREALEN, #CARDSV, AREAVLEN, #CARDSQ, SIZEREG, SIZEV, SIZELV, SIZELQ, and AREAQLEN.

System action: The member is processed only to the specified limit. The rest of the information is skipped.

User response: Increase the value of the relevant parameters, rerun job ASMLINK, and rerun job CADJOUTP.

CTDCAD14E DJDEPARM MEMBER *memname* IS OUT OF SPACE. PROCESSING NEXT REPORT

Explanation: Report member definition *memname* contains more lines than specified in main conversion routine CTDSDJDE, constants #CARDS, AREALEN, #CARDSV, AREAVLEN, #CARDSQ, SIZEREG, SIZEV, SIZELV, SIZELQ and AREAQLEN.

System action: The member is processed only to the specified limit. The rest of the information is skipped.

User response: Increase the value of the relevant parameters, rerun job ASMLINK, and rerun job CADJDJDE.

Appendix C

Building a CONTROL-D Recipient Tree

Use utility CTDBLDTR to create or modify the CONTROL-D Recipient Tree. This utility uses input from two sources: a report (referenced by DD statement REPORT) and a set of instructions (referenced by DD statement SYSIN) specifying how the data in the report is used to create users in the Recipient Tree.

This utility produces a listing (referenced by DD statement SYSPRINT) summarizing the structure (input supplied by the user in SYSIN) and the Recipient Tree (referenced by DD statement TREE). TREE is a member of a partitioned dataset. If TREE is an empty member, the utility creates the Recipient Tree. If TREE contains an existing Recipient Tree, the utility modifies it.

The utility scans each line of the REPORT input and processes it according to the specifications included in the SYSIN data.

For sample JCL programs to execute utility CTDBLDTR, see member BLDTREE in the SOURCE library.

Defining the Levels

The instruction syntax for building the Recipient Tree is as follows:

Figure 3 Instruction Syntax for Building the Recipient Tree

```
LEVEL=xx
USER -
POS=n
LENGTH=n
{    DEFAULT=ccc }
{       POS=n
LENGTH=n
{    DEFAULT=ccc } }
.
.
.
{       POS=n
LENGTH=n
{    DEFAULT=ccc } }
{ PARENT -
PLEVEL=xx
POS=n
LENGTH=n
{    DEFAULT=ccc }
{    TRANSLATE=tabledd } }
{ ADDRESS -
POS=n
LENGTH=n
{    DEFAULT=ccc } }
{ SYNONYM -
POS=n
LENGTH=n
{    DEFAULT=ccc } }
{ SYNONYM -
POS=n
LENGTH=n
{    DEFAULT=ccc } }
END
```

Each LEVEL command can contain four types of paragraphs:

Table 16 LEVEL Command Paragraphs

Paragraph	Description
USER	Instructions to construct the user name. Mandatory.
PARENT	Instructions to construct the parent name. Optional.
ADDRESS	Instructions to construct the address text. Optional.
SYNONYM	Instructions to construct synonyms. Optional. Can be used more than once to construct more than one synonym for each user.

Parameters define how to process the paragraph. These parameters are repeated for the same paragraph if the data to be constructed consists of data contained in more than one string in the report line.

The use of these parameters is as follows:

Table 17 LEVEL Command Parameters

Parameter	Description
POS=*n*	Starting character position of the string in the input report. A value of 0 (zero) indicates that the default value is used. Data from the report is not used for this parameter. The POS value is relative to the first print column of the report (that is, for the first column of the report, POS=1) and does not include print control characters or variable record length values.
LEN=*n*	Length of the data extracted from the input report, starting from the character position specified in POS or from the default value if POS=0.
DEFAULT=*string*	The default value. Optional. Any position in the field of length LEN from the report that is blank is replaced by the corresponding character from parameter DEFAULT.

The combined total length of the data constructed from all the repetitions of the parameters, for each paragraph, must not exceed the number of characters shown in the table below:

Table 18 Maximum Number of Characters for Paragraph Types

# Characters	Paragraph Type
8	USER
8	PARENT
52	ADDRESS
20	SYNONYM

The utility constructs the users based on these definitions and searches the Recipient Tree to see if the constructed user is already defined. If the user is not found, the utility adds the user. If the user is found, the utility updates the Recipient Tree. This utility is especially useful if the only changes required are the addition of synonyms.

Additional Considerations for the PARENT Paragraph

The PARENT paragraph has two additional parameters:

- PLEVEL=*nn*
  Mandatory. Level at which the parent is located.

- TRANSLATE=*tabledd*
  Optional. DD name referencing a file containing the USER/PARENT correspondence.

If the parent of a user cannot be identified from the data on the report line, but can be determined from the user name, a file containing a table relating parent names to user names is supplied.

Each line in the external table is in the format USER=*usermask* PARENT=*parent*

In the *usermask* field, masking characters have the following meaning:

- *—Matches any number of consecutive characters.

- ?—Matches any one character.

Examples

- USER=ABC*D matches users ABC123D, ABC12D, and ABCXD.

- USER=ABC?D only matches user ABCXD from the above set.

TREE Construction Example

Suppose that the SYSIN file contains the following statements:

Figure 4 TREE Construction Example

```
LEVEL=20
USER -
POS=0
LENGTH=1
DEFAULT=L
POS=10
LENGTH=2
DEFAULT=03
PARENT -
PLEVEL=10
POS=0
LENGTH=6
DEFAULT=CDTREE
LEVEL=30
PARENT -
PLEVEL=20
POS=0
DEFAULT=L
LENGTH=1
POS=10
LENGTH=2
TRANSLATE=TABLE1
USER -
POS=0
LENGTH=1
DEFAULT=L
POS=1
LENGTH=4
DEFAULT=CKJ
SYNONYM -
POS=0
LENGTH=4
DEFAULT=USER
POS=1
LENGTH=4
SYNONYM -
POS=0
LENGTH=5
DEFAULT=SYN1-
POS=1
LENGTH=4
```

The Recipient Tree is constructed as follows:

For each line in file REPORT:

1. For a user at level 20 with the name Lxy, where xy are the contents of columns 10 and 11 in the report line:

    - If column 10 is blank, the name is ‘L0y’
    - If column 11 is blank, the name is ‘Lx3’
    - If both are blank, the name is ‘L03’

2. The parent of this user is at level 10 with the name CDTREE.

3. For a user at level 30 with the name Labcd, where abcd are the contents of columns 4 through 7 in the report line:

    Default CKJ is used in a manner similar to default 03 in item 1 above.

4. An attempt is made to determine a level 20 parent from the value ‘L’ plus the contents of columns 10 and 11. If this does not succeed, then the file referenced by DDNAME TABLE1 is scanned line by line until a match is found for the user name and the parent name are taken from the file.

    Two synonyms are created: ‘USERabcd’ and ‘SYN1-abcd’, where abcd represents the contents of columns 1 through 4 in the report line.

Index

Notes